ABSTRACT

Cyber threats, including SQL Injection attacks, are some of the most prevalent and harmful risks to web applications, as input validation weaknesses allow attackers to steal, alter, or delete database data. This research uses the penetration testing method to examine SQL Injection vulnerabilities in web applications. This study differs from previous research, which typically uses simulations and directly tests real systems using two SQL Injection inferential techniques Boolean based blinda and time based blind. To enhance the detection and exploitation of vulnerabilities, SQLMap was configured using advanced parameters such as –level, --risk, and –crawl. The results show that the use of outdated hash algorithms like MD5 and poor input validation make the system highly vulnerable. This research present real exploitation scenarios and offers suggestions mfor technical mitigation based on direct exploitation evidence.The scanning process found 11 vulnerabilities with Nikto and 16 vulnerabilities with OWASP ZAP. These results help developers, administrators, and cybersecurity practitioners maintain web applications in the public sector environment.

Keywords: Penetration testing, SQL Injection, SQLMap, Boolean-Based Blind, Time-Based Blind, Web Security.

ABSTRAK

Ancaman siber termasuk serangan *SQL Injection*, merupakan salah satu ancaman paling umum dan berbahaya terhadap aplikasi web, dikarenakan kelemahan validasi input memungkinkan penyerang dapat mencuri, mengubah, atau menghapus data database. Penelitian ini menggunakan metode *penetration testing* untuk memeriksa kerentanan *SQL Injection* pada aplikasi web, penelitian ini berbeda dari penelitian sebelumnya yang biasanya menggunakan simulasi dan langsung menguji sistem nyata dengan menggunakan dua teknik inferensial *SQL Injection* yaitu *Boolean based blind* dan *time based blind*. Meningkatkan deteksi dan eksploitasi kerentanan, SQLMap dikonfigurasi menggunakan parameter lanjutan seperti `–level`, `--risk`, dan `–crawl`. Hasil menunjukkan bahwa penggunaan algoritma hash MD5 dan validasi input pada web yang tidak ada membuat sistem sangat rentan. Penelitian ini menampilkan skenario eksploitasi nyata dan menawarkan saran untuk mitigasi teknis yang didasarkan pada bukti eksploitasi langsung. Proses *scanning* menemukan 11 kerentanan dengan Nikto dan 16 kerentanan dengan OWASP ZAP. Hasil ini membantu pengembang, administrator dan praktisi keamanan siber mempertahankan aplikasi web di lingkungan sektor publik.

Kata Kunci: *Penetration Testing, SQL Injection, SQLMap, Boolean-Based Blind, Time-Based Blind*, Keamanan Web.